## Architecture du système

La conception du moteur *2D Game Skull* se base sur l’architecture à 3 couches suivante :

***Gameplay***

***Engine***

***API***

Les flèches représentent des relations de dépendances (donc d’obsolescences : la modification d’une couche de base risque d’entraîner la modification de la couche dépendante).

La couche *API* nous est fournie par le *Framework Qt*, il nous reste donc à implémenter les deux couches supérieures, *Engine* et *Gameplay*.

## Ligne à suivre

Notre conception se base autant que possible sur :

* L’utilisation optimale de la couche *API*, afin de ne pas « réinventer la roue » pour les procédures déjà fournies par cette couche primaire,
* Le respect des principes de conception orientée objets (en particuliers les cinq principes *SOLID*),
* Une utilisation –aussi fréquente que possible- des patrons de conception, afin de produire une application de qualité, robuste, et assurer à la fois la *clarté* et la *maintenabilité* du code source.

## Diagramme global des classes participantes

Dans cette section, nous donnons le diagramme des classes qui sert de base à notre conception.

Ce diagramme a été allégé des méthodes et attributs, ainsi que des patrons utilisés, pour ne laisser que le squelette global des classes et de leurs interactions.

*N.B.* : le lecteur trouvera les détails omis dans les sections suivantes de ce document, qui offrent une analyse détaillée de chacune des classes présentées ici.

